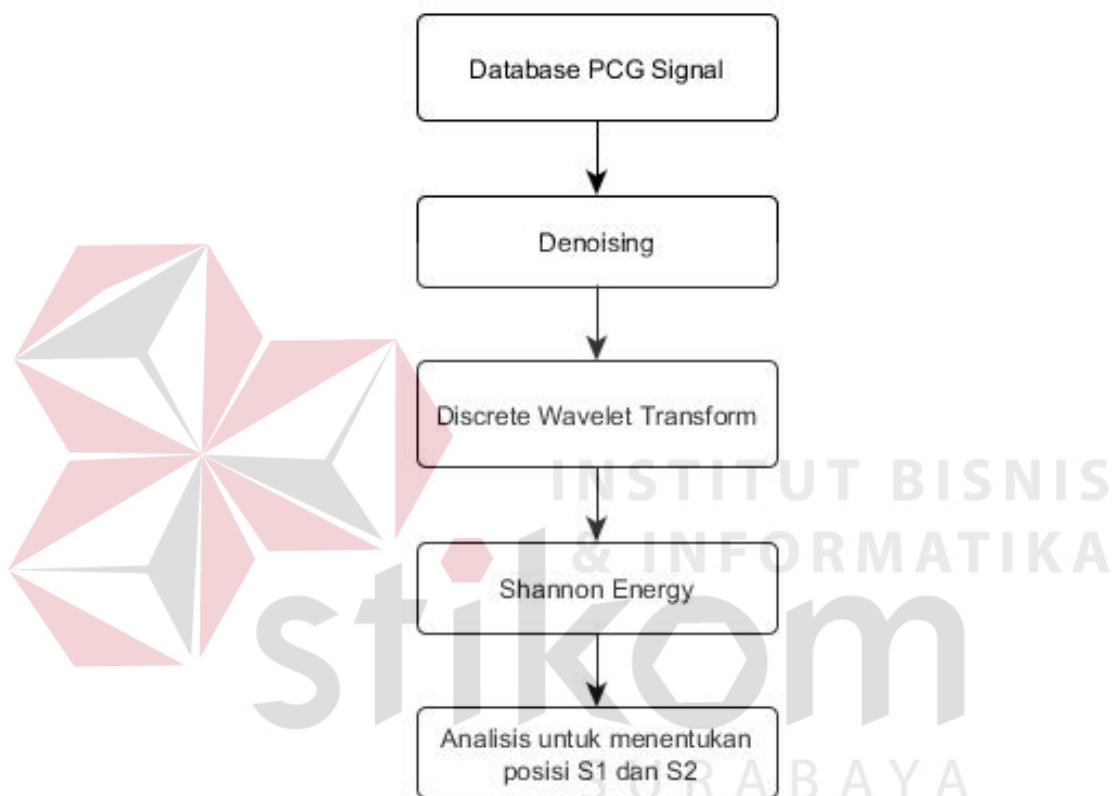


## **BAB III**

### **METODE PENELITIAN DAN PERANCANGAN SISTEM**

#### **3.1 Metode Penelitian**

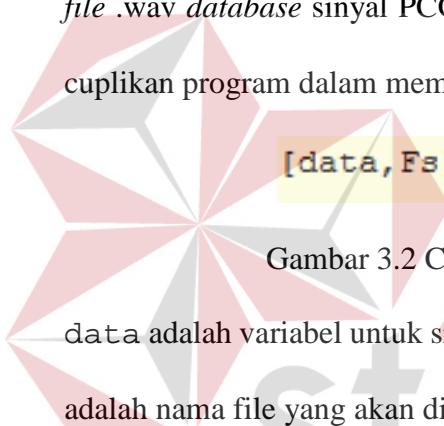


Gambar 3.1. Diagram Blok Rancangan Penelitian

Metode penelitian yang digunakan meliputi studi kepustakaan dan penelitian laboratorium. Studi kepustakaan dilakukan untuk mencari teori atau informasi dari buku, jurnal, dan artikel-artikel yang berkaitan dengan permasalahan, terutama metode penelitian yang akan digunakan. Pada Gambar 3.1 diagram blok rancangan penelitian akan dijelaskan sebagai berikut :

### 3.2 Database PCG Signal

Penelitian ini menggunakan data *real* yang didapat dari *database* hasil penelitian Anggi Citra Ekinasti tahun 2016 yang menggunakan *stethoscope* digital dari *thinklabs one*. Kondisi masing-masing subjek telah di uji melalui uji laboratorium bahwa kondisi jantung subjek adalah jantung normal. Lama pengambilan data pada satu subjek selama 5 detik. Frekuensi sampling yang digunakan dalam penelitian ini adalah 8kHz, 44,1kHz, dan 48kHz. Proses membaca *file .wav* *database* sinyal PCG pada matlab menggunakan *function audioread*, cuplikan program dalam membaca *file* dapat dilihat pada Gambar 3.2.



```
[data, Fs] = audioread('1_2_8.wav');
```

Gambar 3.2 Cuplikan Program Proses Membaca *File .wav*

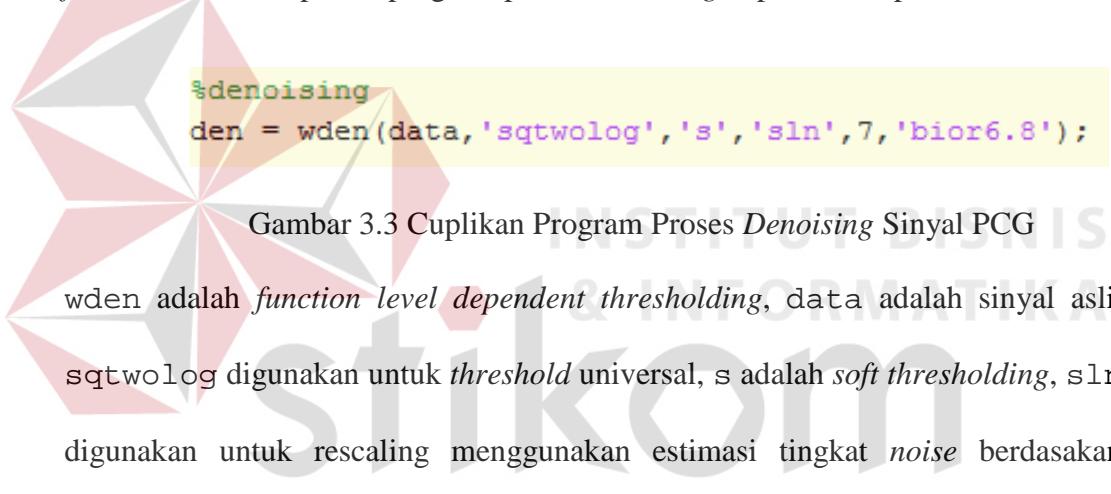
*data* adalah variabel untuk sinyal asli, *Fs* adalah frekuensi sampling, *filename* adalah nama file yang akan dibaca.

### 3.3 Denoising

*Denoising* digunakan untuk menghapus data sinyal yang tidak diperlukan dengan membandingkan nilai *threshold* dan setiap koefisien. Metode *Denoising* yang digunakan adalah *soft thresholding*, dimana metode ini akan membuat nilai yang berada antara  $-T < X < T$  menjadi perlahan menuju 0, sedangkan nilai yang lebih dari  $T$  telah diubah untuk mendekati axis X.

Pada penelitian ini metode yang digunakan adalah *soft thresholding*, menurut Deddy Mohammad Firdaus metode *soft thresholding* memiliki nilai MSE

kecil. *Thresholding rules* yang digunakan adalah *level dependent thresholding*. *Level dependent thresholding* mencari nilai *threshold* berdasarkan level resolusi / level dekomposisi. Penelitian ini level resolusi yang digunakan adalah 7 level dikarenakan sinyal yang diolah memiliki frekuensi sampling sebesar 8kHz. Pada frekuensi sampling 44,1kHz dan 48kHz level resolusi yang digunakan adalah 9 level. Pada penelitian ini penggunaan metode *level dependent thresholding* digunakan *function* pada matlab. Proses *denoising* pada Matlab menggunakan *function* *wden*, cuplikan program proses *denoising* dapat dilihat pada Gambar 3.3.



```
%denoising
den = wden(data, 'sqtwolog', 's', 'sln', 7, 'bior6.8');
```

Gambar 3.3 Cuplikan Program Proses *Denoising* Sinyal PCG

*wden* adalah *function* *level dependent thresholding*, *data* adalah sinyal asli, *sqtwolog* digunakan untuk *threshold* universal, *s* adalah *soft thresholding*, *sln* digunakan untuk rescaling menggunakan estimasi tingkat *noise* berdasarkan koefisien tingkat awal, *level* untuk tingkatan level, *w* adalah *mother wavelet* yang digunakan.

### 3.4 Discrete Wavelet Transform

*Discrete Wavelet Transform* digunakan untuk mendekomposisikan sinyal masukan PCG ke dalam bentuk gelombang sesuai dengan *Mother Wavelet* yang digunakan, dekomposisi dilakukan dengan memisahkan sinyal masukan ke dalam frekuensi rendah dan frekuensi tinggi, hasil dari dekomposisi adalah komponen *approximation* dan komponen *detail* (Sundararajan, 2015).

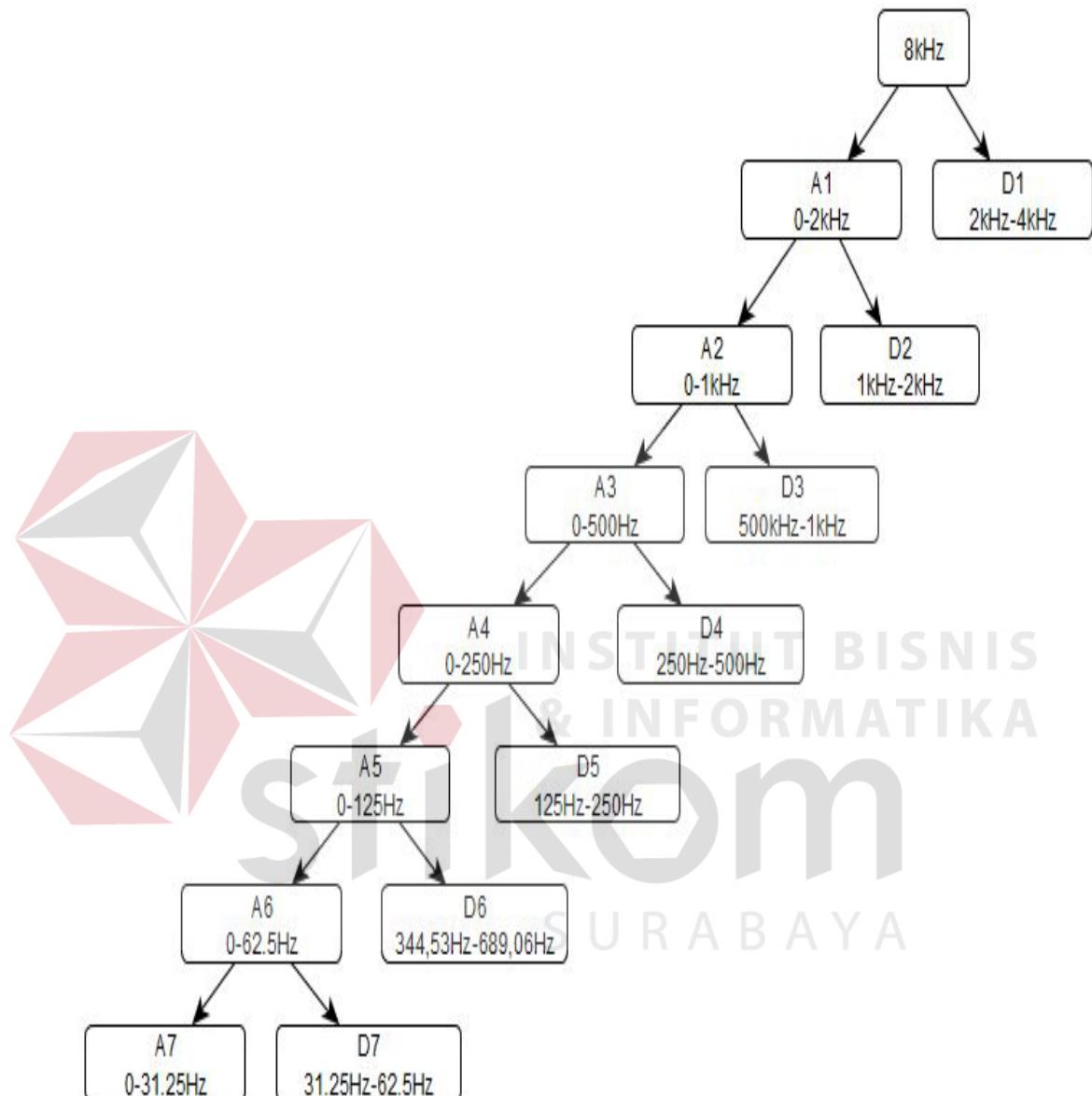
Penelitian ini dipengaruhi beberapa parameter yaitu sinyal PCG dari setiap subjek, frekuensi sampling, *Mother Wavelet*, dan level dekomposisi. Sinyal PCG akan di dekomposisikan menjadi A yang merupakan aproksimasi dan D yang merupakan detail, serta akan di dekomposisikan sesuai dengan frekuensi yang memenuhi frekuensi S1 dan S2. Frekuensi sampling 8kHz akan di dekomposisikan sebanyak 7 tingkat yang dapat dilihat pada Gambar 3.5, frekuensi sampling 44,1kHz dan 48kHz akan di dekomposisikan sebanyak 9 tingkat yang dapat dilihat pada Gambar 3.6 dan Gambar 3.7.

Analisis transformasi *wavelet* diskrit dilakukan dengan mendekomposisi sinyal PCG menggunakan Matlab, untuk mendekomposisikan sinyal satu dimensi maka digunakan fungsi *wavedec*, cuplikan program proses dekomposisi dapat dilihat pada Gambar 3.4.

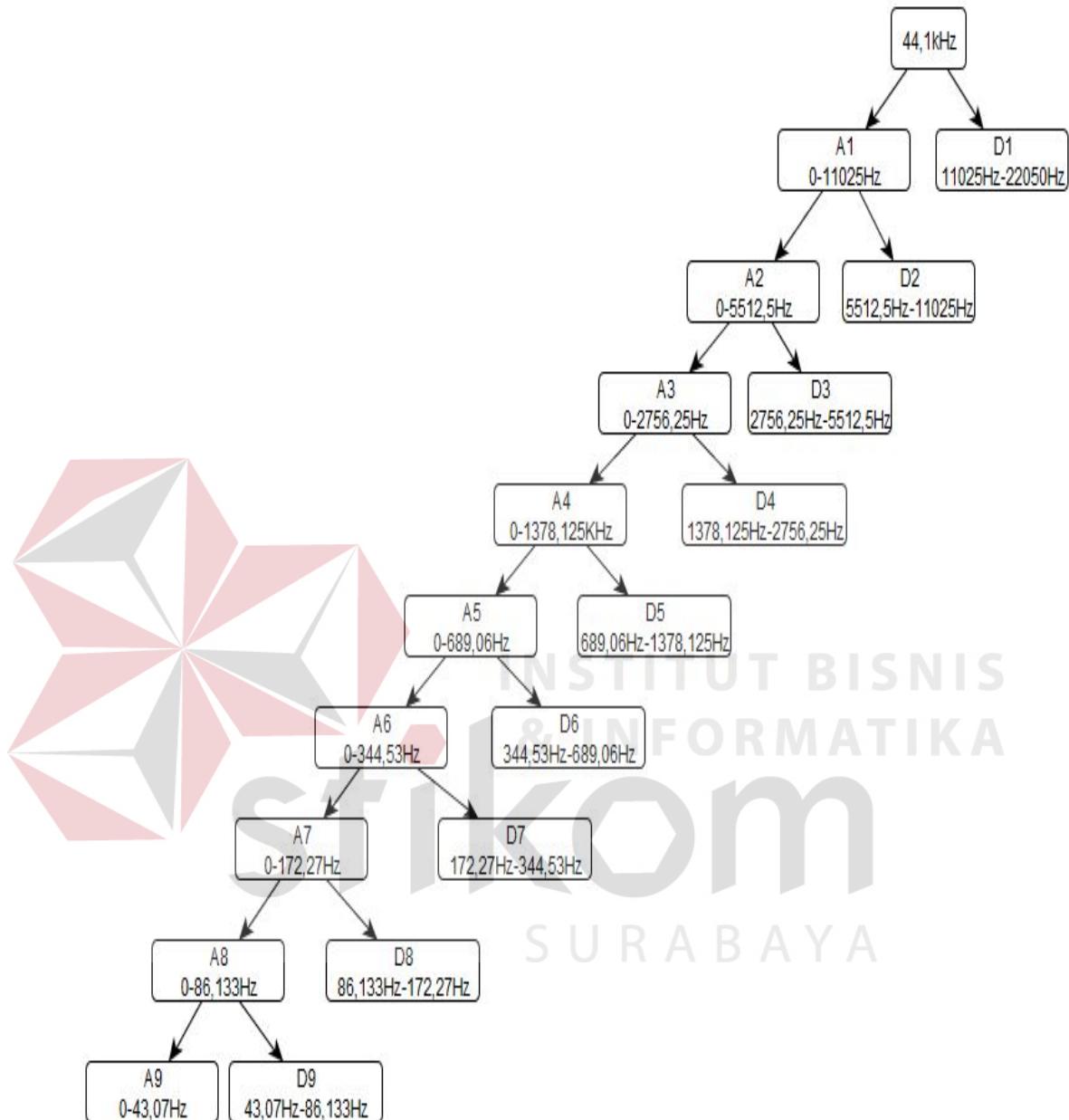
```
%decomposition
n=7;
w='bior6.8';
[C,L]=wavedec(data,n,w);
ca7=appcoef(C,L,w,7);
[cd1,cd2,cd3,cd4,cd5,cd6,cd7]=detcoef(C,L,[1,2,3,4,5,6,7]);
```

Gambar 3.4 Cuplikan Program Proses Dekomposisi

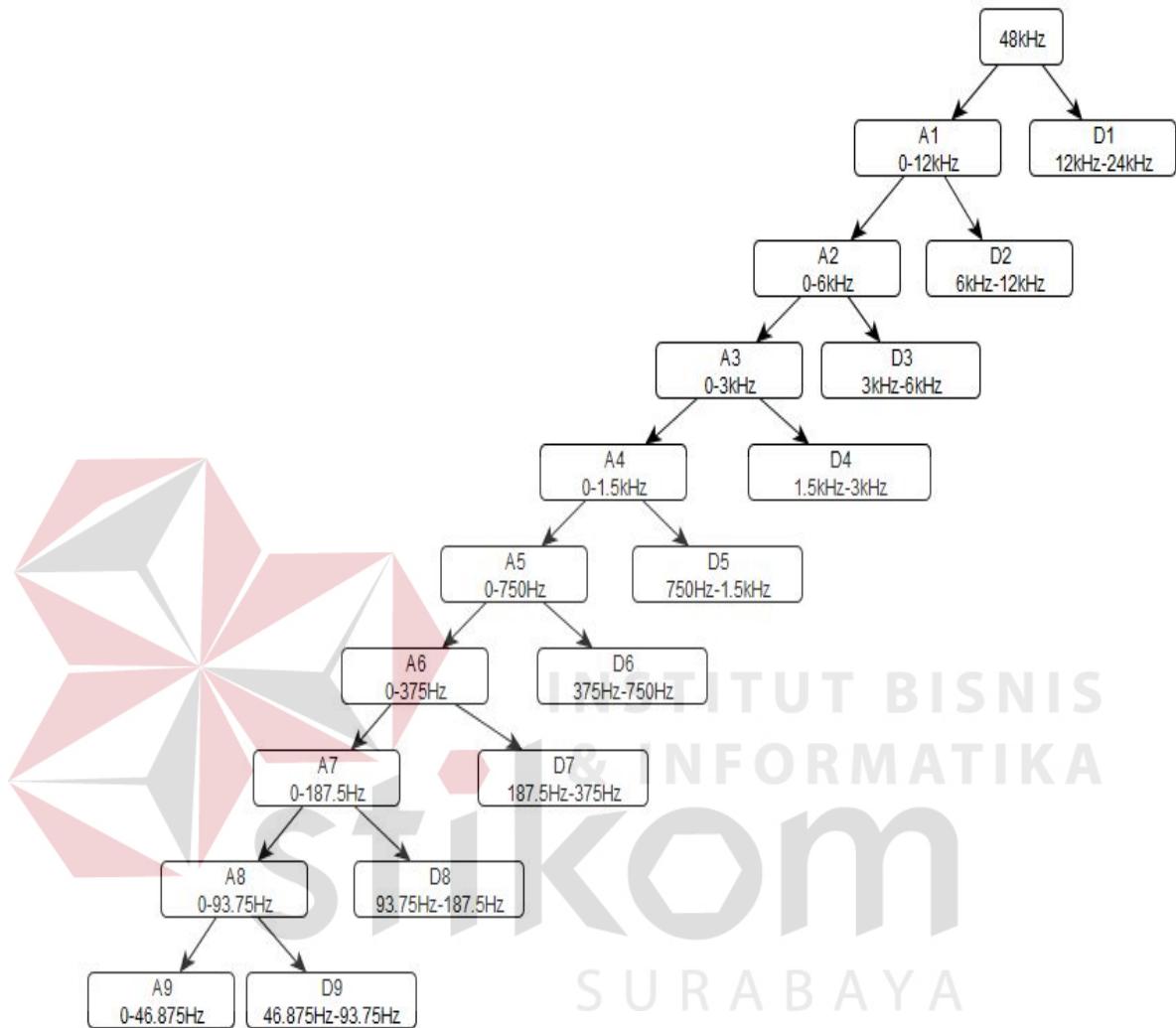
Fungsi *wavedec* ('x', N, 'wname') pada matlab memiliki parameter *input* x untuk sinyal masukan, N untuk tingkatan *level*, dan wname untuk *mother wavelet*, sedangkan parameter *output*-nya adalah hasil dekomposisi dan panjang dari setiap komponen dapat dilihat pada Gambar 3.8.



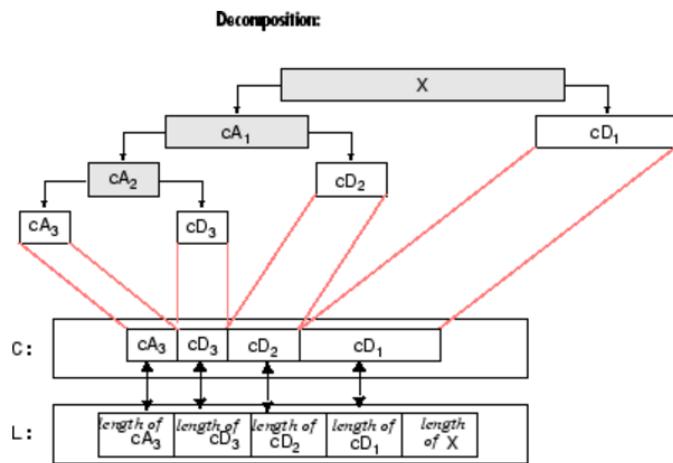
Gambar 3.5 Dekomposisi 7 Tingkat pada Frekuensi Sampling 8kHz



Gambar 3.6 Dekomposisi 9 Tingkat pada Frekuensi Sampling 44,1kHz



Gambar 3.7 Dekomposisi 9 Tingkat pada Frekuensi Sampling 48kHz



Gambar 3.8 Dekomposisi Wavelet Diskrit 1D. (Matlab)

### 3.5 Perhitungan *Shannon Energy*

Proses ini bertujuan untuk mendapatkan nilai *Shannon Energy* beserta *Shannon Envelope*. Sinyal asli sebelum di hitung menggunakan *Shannon Energy* harus di normalisasi terlebih dahulu. Pada penelitian ini normalisasi dilakukan setelah *denoising*. Berikut adalah cuplikan program pada matlab untuk normalisasi sinyal PCG yang dapat dilihat pada Gambar 3.9

```
%normalisasi sinyal pcg
pcg_norm = den/abs(max(den));
```

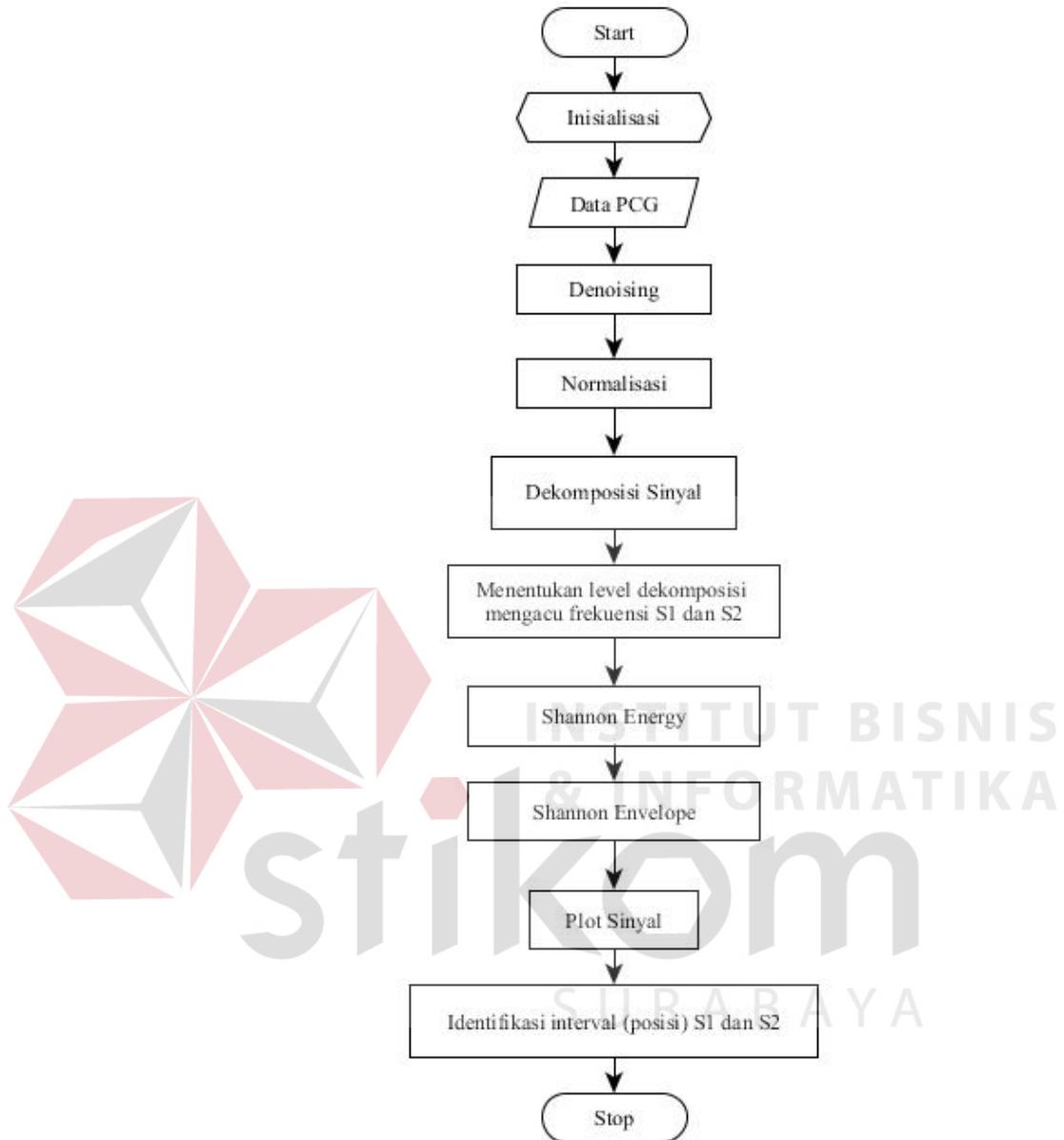
Gambar 3.9 Cuplikan Program Proses Normalisasi Sinyal PCG

Setelah sinyal PCG di normalisasi, sinyal di dekomposisi menurut frekuensi sampling serta menentukan level dekomposisi dengan acuan frekuensi sinyal S1 dan S2. Hasil dekomposisi tersebut di dapat 4 koefisien yang kemungkinan mendekati frekuensi S1 dan S2. Pada frekuensi sampling 8kHz di dapat koefisien Aproksimasi 4, Aproksimasi 5, Aproksimasi 7, Detail 5 dan Detail 6 sedangkan frekuensi sampling 44,1kHz dan 48kHz di dapat koefisien Aproksimasi 7,

Aproksimasi 9, Detail 7, dan Detail 8. Adapun *flowchart* program yang digunakan untuk menghitung *Shannon Energy* akan dijelaskan pada Gambar 3.10.

Gambar 3.10 merupakan gambar *flowchart* dari program perhitungan *Shannon Energy* dan *Shannon Envelope*

1. Inisialisasi variabel yang akan digunakan untuk proses perhitungan *Shannon Energy* dan *Shannon Envelope*
2. *Input* data sinyal PCG
3. Proses *denoising* pada sinyal PCG
4. Proses normalisasi pada sinyal PCG
5. Proses dekomposisi pada sinyal PCG
6. Menentukan level dekomposisi yang mengacu pada frekuensi sinyal S1 dan S2. Pada frekuensi sampling 8kHz koefisien hasil dekomposisi yang dipilih adalah Aproksimasi 4, Aproksimasi 5, Aproksimasi 7, Detail 5, dan Detail 6. Pada frekuensi sampling 44,1kHz dan 48kHz koefisien hasil dekomposisi yang dipilih adalah Aproksimasi 7, Aproksimasi 9, Detail 7, dan Detail 8. Koefisien dekomposisi yang dipilih berdasarkan frekuensi yang mendekati rentang frekuensi sinyal S1 dan S2.
7. Perhitungan *Shannon Energy* pada koefisien dekomposisi yang di pilih, kemudian menghitung *Shannon Envelope*
8. Nilai *Shannon Envelope* akan di *plot*
9. Mengidentifikasi interval waktu posisi S1 dan S2



Gambar 3.10 *Flowchart* Perhitungan *Shannon Energy*